A Study on Hand Gesture Recognition Technique

A THESIS SUBMITTED IN PARTIAL FULFILLMENT

OF THE REQUIREMENTS FOR THE DEGREE OF

Master of Technology

in

Telematics and Signal Processing

By

SANJAY MEENA

Roll No: 209EC1111

Department of Electronics and Communication Engineering

National Institute Of Technology, Rourkela

Orissa 769 008, INDIA

2011

A Study on Hand Gesture Recognition Technique

A THESIS SUBMITTED IN PARTIAL FULFILLMENT

OF THE REQUIREMENTS FOR THE DEGREE OF

Master of Technology

in

Telematics and Signal Processing

By

SANJAY MEENA

Roll No: 209EC1111

Under the Guidance of

Dr. Samit Ari
Assistant Professor

Department of Electronics and Communication Engineering

National Institute Of Technology, Rourkela

Orissa 769 008, INDIA

2011

Dedicated to

To My Parents and my friends

NATIONAL INSTITUTE OF TECHNOLOGY
ROURKELA

CERTIFICATE

This is to certify that the thesis titled **“A Study on Hand Gesture Recognition Technique”** submitted by Mr. **Sanjay Meena** in partial fulfillment of the requirements for the award of Master of Technology degree Electronics and Communication Engineering with specialization in “Telematics and Signal Processing” during session 2009-2011 at National Institute Of Technology, Rourkela is an authentic work by his under my supervision and guidance.

To the best of my knowledge, the matter embodied in the thesis has not been submitted to any other university / institute for the award of any Degree or Diploma.

Date:

Dr. Samit Ari
Assistant Professor
Dept. of Electronics and Comm. Engineering
National Institute of Technology
Rourkela-769008

Acknowledgement

I would like to express my gratitude to my supervisor **Prof. Samit Ari** for his guidance, advice and constant support throughout my thesis work. I would like to thank him for being my advisor here at National Institute of Technology, Rourkela.

Next, I want to express my respects to Prof. S.K. Patra, Prof. K. K. Mahapatra, Prof. S. Meher, Prof. S. K. Behera, Prof. Poonam Singh, Prof. A. K. Sahoo, Prof. D. P. Acharya, prof. S.K. Das and Prof. N. V. L. N. Murty for teaching me and also helping me how to learn. They have been great sources of inspiration to me and I thank them from the bottom of my heart.

I would like to thank all faculty members and staff of the Department of Electronics and Communication Engineering, N.I.T. Rourkela for their generous help in various ways for the completion of this thesis.

I would like to thank all my friends and especially my classmates for all the thoughtful and mind stimulating discussions we had, which prompted us to think beyond the obvious. I've enjoyed their companionship so much during my stay at NIT, Rourkela.

I am especially indebted to my parents for their love, sacrifice, and support and would like to thank my parents for raising me in a way to believe that I can achieve anything in life with hard work and dedication.

Date:

Place:

Sanjay Meena

Roll No: 209EC1111

Dept of ECE, NIT, Rourkela

ABSTARCT

Hand gesture recognition system can be used for interfacing between computer and human using hand gesture. This work presents a technique for a human computer interface through hand gesture recognition that is able to recognize 25 static gestures from the American Sign Language hand alphabet. The objective of this thesis is to develop an algorithm for recognition of hand gestures with reasonable accuracy.

The segmentation of gray scale image of a hand gesture is performed using Otsu thresholding algorithm. Otsu algorithm treats any segmentation problem as classification problem. Total image level is divided into two classes one is hand and other is background. The optimal threshold value is determined by computing the ratio between class variance and total class variance. A morphological filtering method is used to effectively remove background and object noise in the segmented image. Morphological method consists of dilation, erosion, opening, and closing operation.

Canny edge detection technique is used to find the boundary of hand gesture in image. A contour tracking algorithm is applied to track the contour in clockwise direction. Contour of a gesture is represented by a Localized Contour Sequence (L.C.S) whose samples are the perpendicular distances between the contour pixels and the chord connecting the end-points of a window centered on the contour pixels.

These extracted features are applied as input to classifier. Linear classifier discriminates the images based on dissimilarity between two images. Multi Class Support Vector Machine (MCSVM) and Least Square Support Vector Machine (LSSVM) is also implemented for the classification purpose. Experimental result shows that 94.2% recognition accuracy is achieved by using linear classifier and 98.6% recognition accuracy is achieved using Multiclass Support Vector machine classifier. Least Square Support Vector Machine (LSSVM) classifier is also used for classification purpose and shows 99.2% recognition accuracy.

LIST OF FIGURE

LIST OF TABLE

CONTENTS

CHAPTER 1

CHAPTER 2

CHAPTER 3

CHAPTER 4

CHAPTER 5

CHAPTER 1

INTRODUCTION

1.1 HUMAN COMPUTER INTERFACE SYSTEM

Computer is used by many people either at their work or in their spare-time. Special input and output devices have been designed over the years with the purpose of easing the communication between computers and humans, the two most known are the keyboard and mouse [1]. Every new device can be seen as an attempt to make the computer more intelligent and making humans able to perform more complicated communication with the computer. This has been possible due to the result oriented efforts made by computer professionals for creating successful human computer interfaces [1]. As the complexities of human needs have turned into many folds and continues to grow so, the need for Complex programming ability and intuitiveness are critical attributes of computer programmers to survive in a competitive environment. The computer programmers have been incredibly successful in easing the communication between computers and human. With the emergence of every new product in the market; it attempts to ease the complexity of jobs performed. For instance, it has helped in facilitating tele operating, robotic use, better human control over complex work systems like cars, planes and monitoring systems. Earlier, Computer programmers were avoiding such kind of complex programs as the focus was more on speed than other modifiable features. However, a shift towards a user friendly environment has driven them to revisit the focus area [1].

The idea is to make computers understand human language and develop a user friendly human computer interfaces (HCI). Making a computer understand speech, facial expressions and human gestures are some steps towards it. Gestures are the non-verbally exchanged information. A person can perform innumerable gestures at a time. Since human gestures are perceived through vision, it is a subject of great interest for computer vision researchers. The project aims to determine human gestures by creating an HCI. Coding of these gestures into machine language demands a complex programming algorithm. An overview of gesture recognition system is given to gain knowledge.

1.2 GESTURES

It is hard to settle on a specific useful definition of gestures due to its wide variety of applications and a statement can only specify a particular domain of gestures. Many researchers had tried to define gestures but their actual meaning is still arbitrary.

Bobick and Wilson [2] have defined gestures as the motion of the body that is intended to communicate with other agents. For a successful communication, a sender and a receiver must have the same set of information for a particular gesture.

As per the context of the project, gesture is defined as an expressive movement of body parts which has a particular message, to be communicated precisely between a sender and a receiver. A gesture is scientifically categorized into two distinctive categories: dynamic and static [1].

A dynamic gesture is intended to change over a period of time whereas a static gesture is observed at the spurt of time. A waving hand means goodbye is an example of dynamic gesture and the stop sign is an example of static gesture. To understand a full message, it is necessary to interpret all the static and dynamic gestures over a period of time. This complex process is called gesture recognition. Gesture recognition is the process of recognizing and interpreting a stream continuous sequential gesture from the given set of input data.

1.3 GESTURE BASED APPLICATIONS

Gesture based applications are broadly classified into two groups on the basis of their purpose: multidirectional control and a symbolic language.

3D Design: CAD (computer aided design) is an HCI which provides a platform for interpretation and manipulation of 3-Dimensional inputs which can be the gestures. Manipulating 3D inputs with a mouse is a time consuming task as the task involves a complicated process of decomposing a six degree freedom task into at least three sequential two degree tasks. Massachuchetttes institute of technology [3] has come up with the 3DRAW technology that uses a pen embedded in polhemus device to track the pen position and orientation in 3D.A 3space sensor is embedded in a flat palette, representing the plane in which the objects rest .The CAD model is moved synchronously with the users gesture movements and objects can thus be rotated and translated in order to view them from all sides as they are being created and altered.

Tele presence: There may raise the need of manual operations in some cases such as system failure or emergency hostile conditions or inaccessible remote areas. Often it is impossible for human operators to be physically present near the machines [4]. Tele presence is that area of technical intelligence which aims to provide physical operation support that maps the operator arm to the robotic arm to carry out the necessary task, for instance the real time ROBOGEST system [5] constructed at University of California, San Diego presents a natural way of controlling an outdoor autonomous vehicle by use of a language of hand gestures [1]. The prospects of tele presence includes space, undersea mission, medicine manufacturing and in maintenance of nuclear power reactors.

Virtual reality: Virtual reality is applied to computer-simulated environments that can simulate physical presence in places in the real world, as well as in imaginary worlds. Most current virtual reality environments are primarily visual experiences, displayed either on a computer screen or through special stereoscopic displays [6]. There are also some simulations include additional sensory information, such as sound through speakers or headphones. Some advanced, haptic systems now include tactile information, generally known as force feedback, in medical and gaming applications.

Sign Language: Sign languages are the most raw and natural form of languages could be dated back to as early as the advent of the human civilization, when the first theories of sign languages appeared in history. It has started even before the emergence of spoken languages. Since then the sign language has evolved and been adopted as an integral part of our day to day communication process. Now, sign languages are being used extensively in international sign use of deaf and dumb, in the world of sports, for religious practices and also at work places [7]. Gestures are one of the first forms of communication when a child learns to express its need for food, warmth and comfort. It enhances the emphasis of spoken language and helps in expressing thoughts and feelings effectively.

A simple gesture with one hand has the same meaning all over the world and means either ’hi’ or ‘goodbye’. Many people travel to foreign countries without knowing the official language of the visited country and still manage to perform communication using gestures and sign language. These examples show that gestures can be considered international and used almost all over the world. In a number of jobs around the world gestures are means of communication [1].

In airports, a predefined set of gestures makes people on the ground able to communicate with the pilots and thereby give directions to the pilots of how to get off and on the run-way and the referee in almost any sport uses gestures to communicate his decisions. In the world of sports gestures are common. The pitcher in baseball receives a series of gestures from the coach to help him in deciding the type of throw he is about to give. Hearing impaired people have over the years developed a gestural language where all defined gestures have an assigned meaning. The language allows them to communicate with each other and the world they live in.

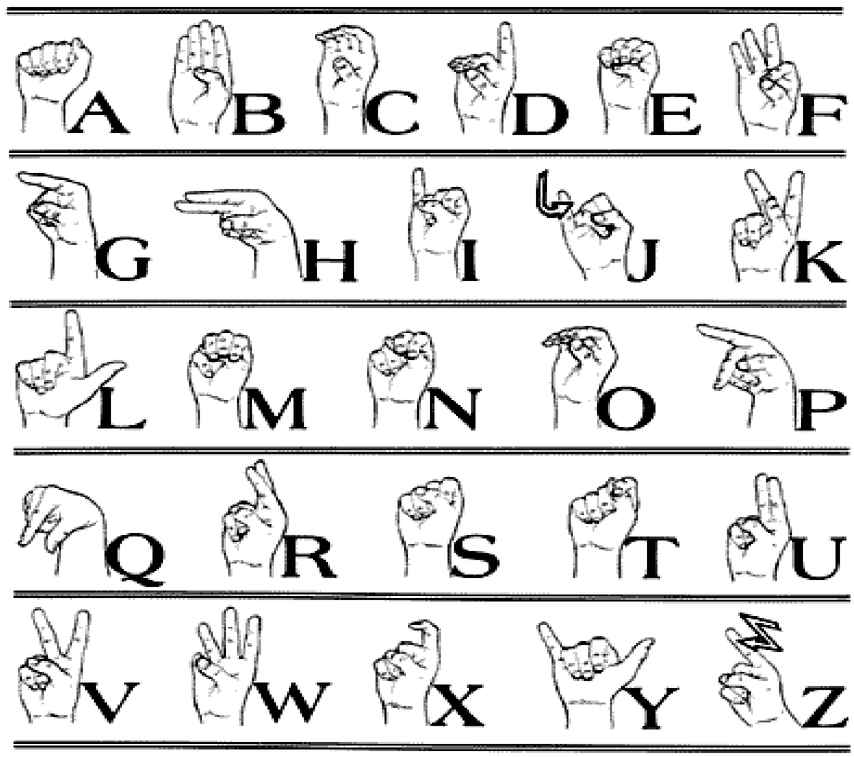


Fig 1.1 American Sign Language [8]

The recognition of gestures representing words and sentences as they do in American and Danish sign language [8] undoubtedly represents the most difficult recognition problem of those applications mentioned before. A functioning sign language recognition system could provide an opportunity for the deaf to communicate with non-signing people without the need for an interpreter. It could be used to generate speech or text making the deaf more independent. Unfortunately there has not been any system with these capabilities so far. In this project our aim is to develop a system which can classify sign language accurately.

1.4 LITERATURE SURVEY

Research has been limited to small scale systems able of recognizing a minimal subset of a full sign language. Christopher Lee and Yangsheng Xu [9] developed a glove-based gesture recognition system that was able to recognize 14 of the letters from the hand alphabet, learn new gestures and able to update the model of each gesture in the system in online mode, with a rate of 10Hz. Over the years advanced glove devices have been designed such as the Sayre Glove, Dexterous Hand Master and PowerGlove [10]. The most successful commercially available glove is by far the VPL DataGlove as shown in figure 1.2

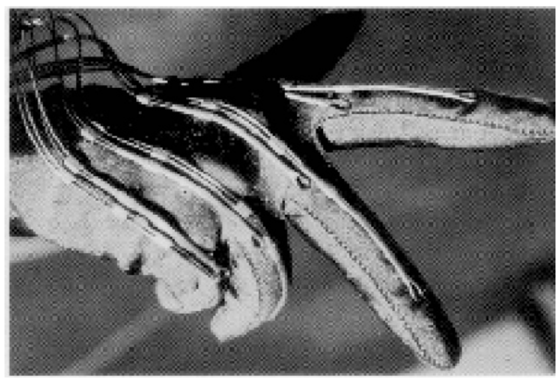


Fig 1.2 VPL data glove [11]

It was developed by Zimmerman [11] during the 1970's. It is based upon patented optical fiber sensors along the back of the fingers. Star-ner and Pentland [3] developed a glove-environment system capable of recognizing 40 signs from the American Sign Language (ASL) with a rate of 5Hz. Hyeon-Kyu Lee and Jin H. Kim [12] presented work on real-time hand-gesture recognition using HMM (Hidden Markov Model) . Kjeldsen and Kendersi [13] devised a technique for doing skin-tone segmentation in HSV space, based on the premise that skin tone in images occupies a connected volume in HSV space. They further developed a system which used a back-propagation neural network to recognize gestures from the segmented hand images. Etsuko Ueda and Yoshio Matsumoto [14] presented a novel technique a hand-pose estimation that can be used

for vision-based human interfaces, in this method, the hand regions are extracted from multiple images obtained by a multiviewpoint camera system, and constructing the “voxel
Model.” Hand pose is estimated. Chan Wah Ng, Surendra Ranganath[15] presented a hand gesture recognition system, they used image furrier descriptor as their prime feature and classified with the help of RBF network . Their system’s overall performance was 90.9%. Claudia Nölker and Helge Ritter [16] presented a hand gesture recognition modal based on recognition of finger tips, in their approach they find full identification of all finger joint angles and based on that a 3D modal of hand is prepared and using neural network.

1.5 SYSTEM OVERVIEW

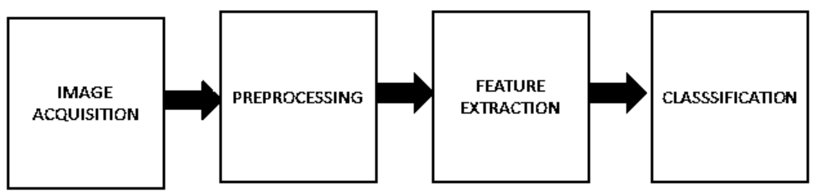


Fig 1.3 Block Diagram of hand gesture recognition system

Vision based analysis, is based on the way human beings perceive information about their surroundings, yet it is probably the most difficult to implement in a satisfactory way. Several different approaches have been tested so far.

- One is to build a three-dimensional model [18] of the human hand. The model is matched to images of the hand by one or more cameras, and parameters corresponding to palm orientation and joint angles are estimated. These parameters are then used to perform gesture classification.
- Second one to capture the image using a camera then extract some feature and those features are used as input in a classification algorithm for classification [19].

In this project we have used second method for modeling the system. In hand gesture recognition system we have taken database from standard hand gesture database, prima database [20]. Segmentation and morphological filtering techniques are applied on images in preprocessing phase then using contour detection we will obtain our prime feature that is Local Contour

Sequence (LCS). This feature is then fed to different classifiers. We have used three classifiers to classify hand gesture images. Linear classifier is our first classifier and then we have used support vector machine (SVM) and least square support vector machine (LSSVM).

1.6 DATABSE DESCRIPTION

In this project all operations are performed on gray scale image .We have taken hand gesture database from [20].The database consist of 25 hand gesture of International sign language. The letter j,z and have been discard for their dynamic content. Gesture ae is produced as it is a static gesture .The system works offline recognition ie. We give test image as input to the system and system tells us which gesture image we have given as input. The system is purely data dependent.

We take gray scale image here for ease of segmentation problem. A uniform black background is placed behind the performer to cover all of the workspace. The user is required to wear a black bandage around the arm reaching from the wrist to the shoulder. By covering the arm in a color similar to the background the segmentation process is fairly straight forward.

A low-cost black and white camera is used to capture the hand gesture performed by performer .it produces 8-bit gray level image. The resolution of grabbed image is 256*248. Each of the gestures/signs is performed in front of a dark background and the user's arm is covered with a similar black piece of cloth, hence easy segmentation of the hand is possible. Each gesture is performed at various scales, translations, and a rotation in the plane parallel to the image-plane [20].There are total 1000 images, 40 images per gesture.


a

ae

b

c

d

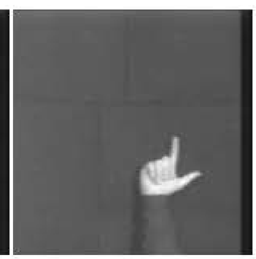
e

f

g

h

I

k

l

m

n

o

p

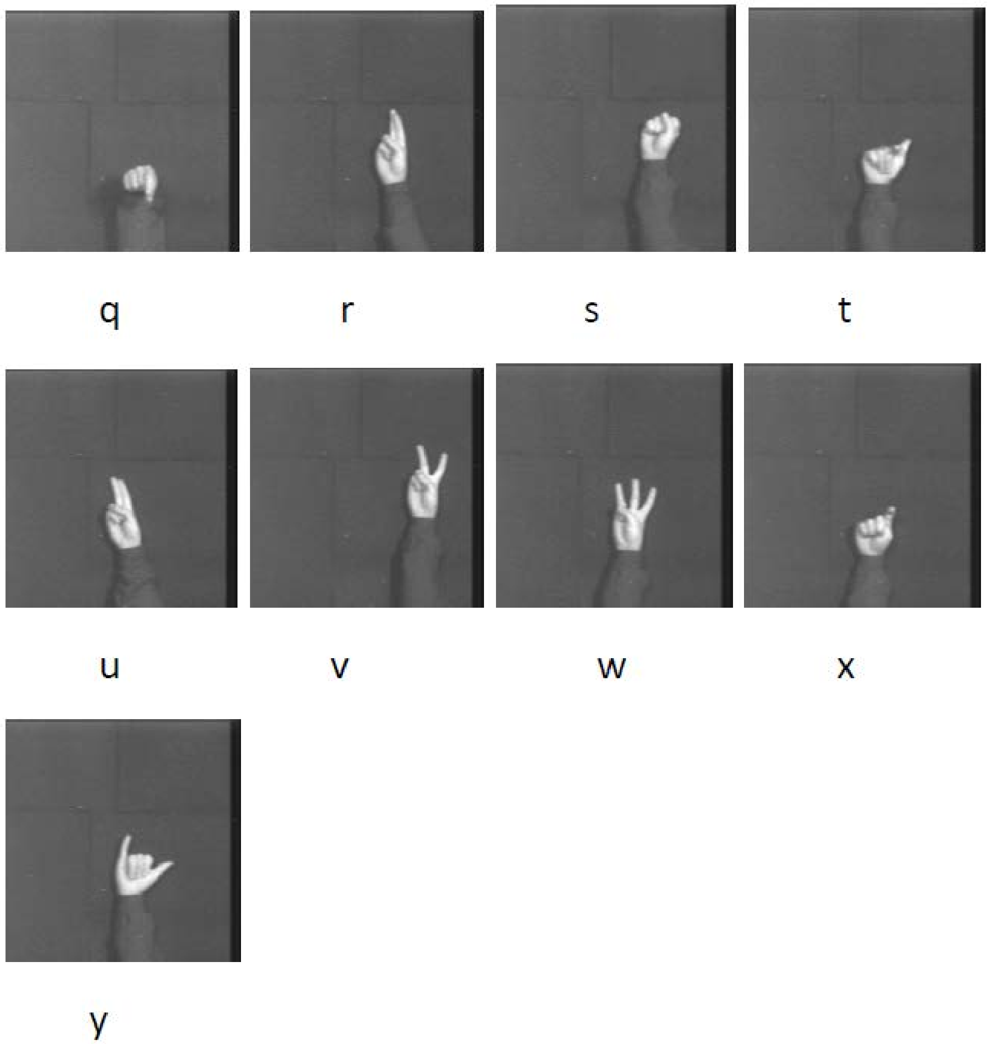


Fig 1.4 Samples of Images from database [18]

1.6 THESIS OUTLINE

In **Chapter 2** Preprocessing of gesture recognition system is described. Preprocessing consist image acquisition, segmentation and morphological filtering methods. We have taken our database from prima database .Otsu algorithm is used for segmentation purpose and gray scale images is converted into binary image consisting hand or background .Morphological filtering techniques are used to remove noises from images so that we can get a smooth contour.

In **Chapter 3** feature extraction methods is described .We have used Local contour sequence as our prime feature. Canny edge detection technique is used to detect the border of hand in image.

A contour tracking is applied to find the contour and pixel in contour is numbered sequentially. Local contour sequence for any arbitrary pixel is calculated as perpendicular distance from the chord connecting end points of window size *w*.

In **chapter 4** we explained different technique of classification of hand gesture using LCS feature linear classifier, Support Vector machine and least square support machine theory

In **chapter 5** we concluded our work and discussed about its future scope.

REFERENCES

[1] Henrik Birk and Thomas Baltzer Moeslund, “Recognizing Gestures From the Hand Alphabet Using Principal Component Analysis”, Master’s Thesis, Laboratory of Image Analysis, Aalborg University, Denmark, 1996.

[2] Andrew Wilson and Aaron Bobick, “Learning visual behavior for gesture analysis,” *In Proceedings of the IEEE Symposium on Computer Vision*, Coral Gables, Florida, pp. 19-21, November 1995.

[3] Thad Starner and Alex Pentland, “Real-time American sign language recognition from video using hidden markov models”,Technical Report No. 375, M.I.T Media Laboratory Perceptual Computing Section, 1995.

[4] Jennifer Schlenzig, Edward Hunter, and Ramesh Jain, “Recursive spatio-temporal analysis: Understanding Gestures”, Technical report, Visual Computing Laboratory, University of San Diego, California, 1995.

[5] Arun Katkere, Edward Hunter, Don Kuramura, Jennifer Schlenzig, Saied Moezzi, and Ramesh Jain, “Robogest: Telepresence using hand gestures”,Technical report, University of California, San Diego, Visual Computing Laboratory, Technical Report No. VCL-94-104, December 1994.

[6] Hank Grant, Chuen-Ki Lai, “simulation modeling with artificial reality technology (smart): an integration of virtual reality simulation modeling” , Proceedings of the Winter Simulation Conference, 1998.

[7] Theodore Brun. “Teckensprks Lexikon”. Bokforlaget Spektra AB, Halmstad, 1974

[8] http://www.happinesspages.com/baby-sign-language-FAQ.html

[9] Christopher Lee and Yangsheng Xu, “Online, interactive learning of gestures for human robot interfaces” Carnegie Mellon University, The Robotics Institute, Pittsburgh, Pennsylvania, USA, 1996

[10] RichardWatson, “Gesture recognition techniques”, Technical report, Trinity College, Department of Computer Science, Dublin, July, Technical Report No. TCD-CS-93-11, 1993

[11] Thomas G. Zimmerman , Jaron Lanier , Chuck Blanchard , Steve Bryson , Young Harvill, “A hand gesture interface device”, SIGCHI/GI Proceedings, conference on Human factors in computing systems and graphics interface, p.189-192, April 05- 09, , Toronto, Ontario, Canada, 1987

[12] Hyeon-Kyu Lee and Jin H. Kim,” An HMM-Based Threshold Model Approach for Gesture Recognition” IEEE *transactions on pattern analysis and machine intelligence*, vol. 21, no. 10, october 1999

[13] Rick Kjeldsen and John Kender,“Finding skin in colour images”, *In Proc. IEEE Int. Conf. on autom. Face and Gesture Recognition*, pages 3 12-3 17, 1996

[14] Etsuko Ueda, Yoshio Matsumoto, Masakazu Imai, Tsukasa Ogasawara. “Hand Pose Estimation for Vision-based Human Interface”, *IEEE Transactions on Industrial Electronics*, Vol. 50, No. 4, pp. 676–684,2003.

[15] Chan Wah Ng, Surendra Ranganath, “Real-time gesture recognition system and application”, Image Vision Comput, 20(13-14): 993-1007 ,2002

[16] Claudia Nölker and Helge Ritter, “Visual Recognition of Continuous Hand Postures”, *IEEE transactions on systems, man, and cybernetics—part c: applications and reviews,* vol. 31, no. 1, February 2001

[17] Bowden and Sarhadi, “Building temporal models for gesture recognition” British Machine Vision Conference, pages 32-41, 2000.

[18] Matthew A. Turk and Alex P. Pentland, “ Face recognition using eigenfaces”, *IEEE Society Conference on Computer Vision and Pattern Recognition*, pages 586–591, Lahaina, Maui, Hawaii, June 3-6 1991.

[19] J. Edward Jackson, “A Users Guide to Principal Components”, Wiley Series in Probability and Mathematical Statistics, A Wiley-Interscience Publication, 1st edition, 1991.

[20] http://www-prima.inrialpes.fr/FGnet/data/12-MoeslundGesture/database.html

CHAPTER 2
PREPROCESSING

2.1 INTRODUCTION

Preprocessing is very much required task to be done in hand gesture recognition system. We have taken prima database [1] which is standard database in gesture recognition. We have taken total 25 signs each sign with 40 images. Preprocessing is applied to images before we can extract features from hand images. Preprocessing consist of two steps

- Segmentation
- Morphological filtering

Segmentation is done to convert gray scale image into binary image so that we can have only two object in image one is hand and other is background. Otsu algorithm [2] is used for segmentation purpose and gray scale images are converted into binary image consisting hand or background. After converting gray scale image into binary image we have to make sure that there is no noise in image so we use morphological filter technique. Morphological techniques consist of four operations: dilation, erosion, opening and closing.

2.2 SEGMENTATION

A very good segmentation is needed to select a adequate threshold of gray level for extract hand from background .*i.e.* there is no part of hand should have background and background also shouldn't have any part of hand. In general, the selection of an appropriate segmentation algorithm depends largely on the type of images and the application areas. The Otsu segmentation algorithm [2] was tested and found to give good segmentation results for the hand gestures and was, therefore, selected .Otsu algorithm is nonparametric and unsupervised method of automatic threshold selection [2].

Let the pixels of a given picture be represented in L gray levels *[1,2,3,……………..,L]* The number of pixels at level i is denoted by n_i and the total number of pixels by $N = n_1 + n_2 + n_3 + n_4 \ldots \ldots . n_L$. Now the probability distribution of pixel is given by [2]

$$p_i = \frac{n_i}{N}, \qquad p_i \geq 0, \sum_{i=1}^{L} p_i = 1 \qquad (2.2.1)$$

Let us suppose we have two classes of pixels one is Ω_0 which is background and Ω_1 is the hand. Ω_0 shows the pixels with level [*1….k*], and Ω_1 shows pixels with level [*k+1….L*].The probability of class occurrence and the class mean levels, respectively, are given by

$$\omega_0 = P(\Omega_0) = \sum_{i=1}^{k} p_i = \omega(k) \tag{2.2.2}$$

$$\omega_1 = P(\Omega_1) = \sum_{i=k+1}^{L} p_i = 1 - \omega(k) \tag{2.2.3}$$

and

$$\mu_0 = \sum_{i=1}^{k} iP(i|\Omega_0) = \sum_{i=1}^{k} ip_i/\omega_0 = \mu(k)/\,\omega(k) \tag{2.2.4}$$

$$\mu_1 = \sum_{i=k+1}^{L} iP(i|\Omega_1) = \sum_{i=k+1}^{L} ip_i/\omega_1 = \frac{\mu_T - \mu(k)}{1-\omega(k)} \tag{2.2.5}$$

where

$$\omega(k) = \sum_{i=1}^{k} p_i \tag{2.2.6}$$

and

$$\mu(k) = \sum_{i=1}^{k} ip_i \tag{2.2.7}$$

These two are *zero*th and the *first* order cumulative moments of the histogram up to k^{th} level and respectively [2]

$$\mu_T = \mu(L) = \sum_{i=1}^{k} ip_i \tag{2.2.8}$$

Here μ_T is total mean level of the original hand image. So we can check for relation for any value of *k*

$$\omega_0\mu_1 + \omega_1\mu_1 = \mu_T, \qquad \omega_0 + \omega_1 = 1$$

The class variance for both class is given by

$$\sigma_0^2 = \sum_{i=1}^{k}(i-\mu_0)^2 P(i|\Omega_0) = \sum_{i=1}^{k}(i-\mu_0)^2\, p_i/\omega_0 \tag{2.2.9}$$

$$\sigma_1^2 = \sum_{i=k+1}^{L}(i-\mu_1)^2 P(i|\Omega_1) = \sum_{i=k+1}^{L}(i-\mu_1)^2\, p_i/\omega_1 \tag{2.2.10}$$

and between classes variance is given by

$$\sigma_B^2 = \omega_0(\mu_0 - \mu_T)^2 + \omega_0(\mu_1 - \mu_T)^2 \qquad (2.2.11)$$

and total class variance is given by

$$\sigma_B^2 = \sum_{i=1}^{L}(i - \mu_T)^2\, p_i \qquad (2.2.11)$$

Now we will find the ratio of between class variance to the total class variance with all value for (*i=1,2,3, …k,k+1….L)* .the optimum threshold k* is determined the value of pixel for which we get the maximum value of σ_B^2/σ_T^2 [3].

$$k^* = max_{1\le k\le L}\, \sigma_B^2/\sigma_T^2 \qquad (2.2.12)$$

Now we set our threshold k* and hand pixel is assigned "*1*" and the background pixels are assigned "*0*" thus we get a binary image.

2.3 MORPHOLOGICAL FILTERING

If we take close look to the segmented image after applying the Otsu algorithm on the original gray scale image we find that the segmentation is not perfectly done. Background may have some 1s which is known as background noise and hand gesture mat have some 0s that is known is gesture noise. These errors can lead to a problem in contour detection of hand gesture so we need to remove these errors. A morphological filtering [4] approach has been applied using sequence of dilation and erosion to obtain a smooth, closed, and complete contour of a gesture.

In the morphological dilation and erosion we apply a rule on a binary image. The value of any given pixel of any given pixel in output image is obtained by allying set of rules on the neighbors in the input image [5]. The dilation and erosion operation on a binary image A and with a structuring element B defined as follow [4].

Dilation: If A and B are sets in the *2-D* integer spaceZ^2, $x = (x_1, x_2)$ and $\emptyset$ is the empty set, then, the dilation of A by B is

$$A \oplus B = \{x | (\hat{B})x \cap A \neq \emptyset\}$$

Where $\hat{B}$ is the reflection of *B*. In dilation process first we obtain the reflection of *B* about its origin and then we shift the reflection by x [3].The condition of dilation of *A* by *B* is set of all x The condition is such that for dilation [6] of *A* by *B* is set off all x displacement such that $\hat{B}$ and *A* overlap at least one nonzero element. Set B is commonly referred to as the structuring element [4]. The value of the output pixel is the maximum value of all the pixels in the input pixel's neighborhoods. In any of the pixels is set to the value *1*, the output pixel is set to *1*.

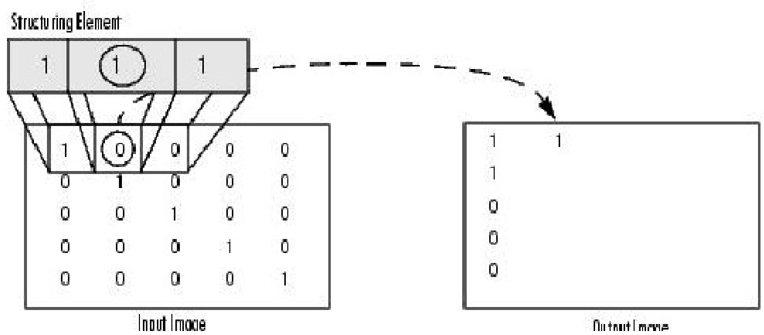


Fig 2.1 dilation Process [4]

Erosion: the erosion of *A* by *B* is

$$A \otimes B = \{x | (B)x \subseteq A\}$$

The erosion of *A* by *B* is the set of all point *x* such that *B*, translated by *A*, is contained in *A* [3]. Thus the value of the output pixel is minimum value of all the pixels in the input pixel's neighborhood. In binary image, if any of the pixels is set to *0*, the output pixel is set to *0* [7].

Opening: The opening of *A* by *B* is obtained by the erosion of *A* by *B*, followed by dilation of the resulting image by *B*.

$$A \circ B = (A \otimes B) \oplus B$$

Opening essentially removes the outer tiny "hairline" leaks [8] and restores the image .the side effect of opening that it round off things so sharp edges start to disappear.

Closing: the closing of set *A* by structuring element *B* is

$$A \bullet B = (A \oplus B) \otimes B$$

The opening of A by B is simply the erosion of A by B followed by a erosion of the result by B. Closing also tends to smooth section of contours but [3], as opposed ,it generally fuses narrow breaks and long thin gulfs, eliminates small holes and fills gaps in the contour [3].

2.4 RESULT

2.4.1 SEGMENTATION RESULT

Segmentation in our proposed hand gesture recognition system is done by Otsu algorithm. The algorithm treats the segmentation of a gray scale image into a binary image as a classification problem in which the two classes (in this case, hand and background) are generated from the set of pixels within the gray scale image [3]. There are total L levels in ray scale image (0-255) Using a threshold T for an image with L gray levels, the image is segmented in two classes$\Omega_0 = (1,2 \ldots\ldots k)$ and $\Omega_0 = (k+1, k+2 \ldots\ldots L)$ The optimum threshold k^* is determined as that value of k which maximizes the ratio of the between-class variance σ_B^2 B to the total variance σ_k^2.after finding the threshold value k hand pixeal were assigned “1” and the background pixels were assigned “0”.Segmentation results are shown below:

Unsegmented image of gesture “a” Segmented image of gesture “a”

Fig 2.2 Segmentation of gray scale gesture image of gesture “a”

Unsegmented image of gesture “b”   Segmented image of gesture “b”

Fig 2.3 Segmentation of gray scale gesture image of gesture “b”

Unsegmented image of gesture “c”   Segmented image of gesture “c”

Fig 2.4 Segmentation of gray scale gesture image of gesture “c”

Unsegmented image of gesture “d”   Segmented image of gesture “d”

Fig2.5 Segmentation of gray scale gesture image of gesture “d”

2.4.2 MORPHOLOGICAL FILTERING RESULT

After finding the segmented image we find that it has some noise so for reduction noise we used morphological filtering operations .Results after morphological operations is given below

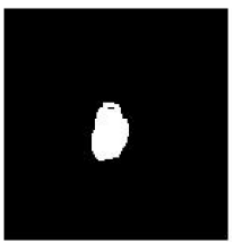
Segmented image of gesture “a”

Morphlogical Filtered image of gesture “a”

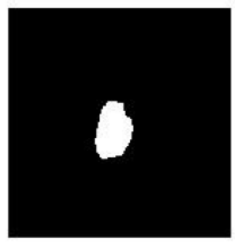
Segmented image of gesture “b”

Morphological Filtered image of gesture “b”

Fig 2.6 morphological Filtered image of gesture “a”and”b”



Segmented image of gesture “c”



Morphological Filtered image of gesture “c”

Segmented image of gesture “d”

Morphlogical Filterd image of gesture “d”

Fig 2.7 Morphological Filtered Gesture “c”and “d”

2.5 CONCLUSION

In this chapter Preprocessing of gesture recognition system is described. Preprocessing consist image acquisition, segmentation and morphological filtering methods. We have taken our database from prima database .Otsu algorithm is used for segmentation purpose and gray scale images is converted into binary image consisting hand or background .Morphological filtering techniques are used to remove noises from images so that we can get a smooth contour.

REFERENCE

[1] http://www-prima.inrialpes.fr/FGnet/data/12-MoeslundGesture/database.html

[2] N.Otsu, “A Threshold Selection Method from Gray-Level Histograms”, *IEEE transactions on systems, man, and cybernetics*, vol. smc-9, no. 1, January 1979.

[3] Lalit Gupta and Suwei Ma “Gesture-Based Interaction and Communication: Automated Classification of Hand Gesture Contours”, *IEEE transactions on systems, man, and cybernetics—part c: applications and reviews*, vol. 31, no. 1, February 2001

[4] E. R. Dougherty, “An Introduction to Morphological Image Processing”, Bellingham, Washington: SPIE Optical Engineering Press, 1992.

[5] L. Gupta and T. Sortrakul, “A Gaussian mixture based image segmentation algorithm,” *Pattern Recognit.*, vol. 31, no. 3, pp. 315–325, 1998.

[6] V. I. Pavlovic, R. Sharma, and T. S. Huang, “Visual interpretation of hand gestures for human computer interaction: A review,” *IEEE Trans. Pattern Anal. Machine Intell.*, vol. 19, pp. 677–694, July 1997.

[7] L. Gupta, T. Sortrakul, A. Charles, and P. Kisatsky, “Robust automatic target recognition using a localized boundary representation,” Pattern Recognit., vol. 28-10, pp. 1587–1598, 1995.

[8] H. Sakoe and S. Chiba, “Dynamic programming optimization for spoken word recognition,” *IEEE Trans. Acoust. Speech Signal Processing*, vol. ASSP-26, pp. 43–49, Feb. 1978.

[9] Vladimir I. Pavlovic, Rajeev Sharma, Thomas S. Huang, "Visual Interpretation of Hand Gestures for Human-Computer Interaction: A Review", *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 19, no. 7, pp. 677-695, July 1997

[10] B. Moghaddam and A. Pentland, “Probabilistic visual learning for object recognition,” *IEEE Trans. Pattern Anal. Machine Intell*., vol. 19, pp. 696–710, July 1997.

CHAPTER 3

FEATURE EXTRACTION

3.1 INTRODUCTION

In this chapter we will discuss the feature extraction for the purpose of gesture recognition. Feature extraction is very important in terms of giving input to a classifier .Our prime feature is local contour sequence (L.C.S) .In feature extraction first we have to find edge of the segmented and morphological filtered image . Canny edge detector algorithm is used to find the edge which leads us to get boundary of hand in image. Then a contour tracking algorithm is applied to track the contour [1].

3.2 CANNY EDGE DETECTOR

In image processing finding edge is fundamental problem because edge defines the boundaries of different objects. Edge can be defined as sudden or strong change in the intercity or we can say sudden jump in intensity from one pixel to other pixel. By finding the edge in any image we are just reducing some amount of data but we are preserving the shape. The Canny edge detection algorithm is known as the optimal edge detector. Canny [2], improved the edge detection by following a list of criteria. The first is low error rate. Low error rate means edges occurring in images should not be missed and that there are NO responses to non-edges [3]. The second criterion is that the edge points be well localized. In other words, the distance between the edge pixels as found by the detector and the actual edge is to be at a minimum. A third criterion is to have only one response to a single edge [3]. This was implemented because the first 2 were not substantial enough to completely eliminate the possibility of multiple responses to an edge . Based on these criteria, the canny edge detector first smoothes the image to eliminate and noise. It then finds the image gradient [2] to highlight regions with high spatial derivatives [3]. The algorithm then tracks along these regions and suppresses any pixel that is not at the maximum (nonmaximum suppression). The gradient array is now further reduced by hysteresis. Hysteresis is used to track along the remaining pixels that have not been suppressed. Hysteresis uses two thresholds and if the magnitude is below the first threshold, it is set to zero (made a nonedge). If the magnitude is above the high threshold, it is made an edge. And if the magnitude is between the 2 thresholds, then it is set to zero unless there is a path from this pixel to a pixel with a gradient above T2 [3].

Step 1: In order to implement the canny edge detector algorithm, a series of steps must be followed. The first step is to filter out any noise in the original image before trying to locate and detect any edges. And because the Gaussian filter can be computed using a simple mask [3], it is used exclusively in the Canny algorithm. Once a suitable mask has been calculated, the Gaussian smoothing can be performed using standard convolution methods [4]. A convolution mask is usually much smaller than the actual image [3]. As a result, the mask is slid over the image, manipulating a square of pixels at a time. The larger the width of the Gaussian mask, the lower is the detector's sensitivity to noise. The localization error in the detected edges also increases slightly as the Gaussian width is increased. Example of a 5*5 Gaussian filter is given below [3]

$$\frac{1}{273}\begin{bmatrix} 1 & 4 & 7 & 4 & 1 \\ 4 & 16 & 26 & 16 & 4 \\ 7 & 26 & 41 & 26 & 7 \\ 4 & 16 & 26 & 16 & 4 \\ 1 & 4 & 7 & 4 & 1 \end{bmatrix}$$

*Fig 3.1 A 5 * 5 Gaussian filter Example[2]*

Step 2: After smoothing the image and eliminating the noise, the next step is to find the edge strength by taking the gradient of the image. The Sobel operator performs a 2-D spatial gradient measurement on an image [3]. Then, the approximate absolute gradient magnitude (edge strength) at each point can be found. The Sobel operator uses a pair of 3x3 convolution masks, one estimating the gradient in the x-direction (columns) and the other estimating the gradient in the y-direction (rows). They are shown below

-1	0	+1
-2	0	+2
-1	0	+1

Gx

+1	+2	+1
0	0	0
-1	-2	-1

Gy

Fig 3.2 Gradient example[2]

$$G = \sqrt{G_x^2 + G_y^2}$$

From this the edge gradient and the direction can be determined [3]

$$\theta = arctan\left(\frac{G_y^2}{G_x^2}\right)$$

step3: Once the edge direction is known, the next step is to relate the edge direction to a direction that can be traced in an image [2]. So if the pixels of a 5x5 image are aligned as follows:

0	0	0	0	0
0	0	0	0	0
0	0	1	0	0
0	0	0	0	0
0	0	0	0	0

*Fig 3.3 Image segment (5*5)*

Then, it can be seen by looking at pixel whose value is"1", there are only four possible directions when describing the surrounding pixels - 0 degrees (in the horizontal direction), 45 degrees (along the positive diagonal), 90 degrees (in the vertical direction), or 135 degrees (along the negative diagonal). So now the edge orientation has to be resolved into one of these four directions.[3]

step4: After the edge directions are known, nonmaximum suppression now has to be applied. Nonmaximum suppression is used to trace along the edge in the edge direction and suppress any pixel value (sets it equal to 0) that is not considered to be an edge. This will give a thin line in the output image.[3]

3.3 LOCALIZED CONTOUR SEQUENCE

After edge detection we get a boundary of hand in image that is our contour of hand image .now a algorithm is applied on the contour to tarck it in clockwise direction and the contour pixel are numberd sequentially [1],[5].first we ran a search in image to find a topmost nonzero i.e contour pixel then numbered the contour in sequential order in clockwise direction from that point.

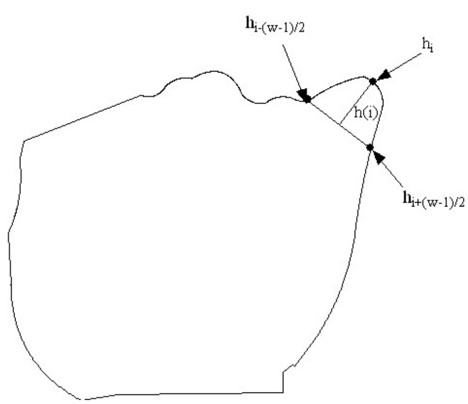


Fig 3.4 computation of LCS of a contour

Let us suppose there are total N points in the above contour then $h_i = (x_i, y_i)$,i=1,2,….*N*, is the ith contour pixel . the ith sample $h(i)$ sample of LCS of the gesture is obtained by computing the perpendicular Euclidean distence between h_i and chord connecting the end-points $h_{[i-(w-1)/2]}$ and $h_{[i+(w-1)/2]}$ of a window of size w boundary pixel centerd on $h_i[i]$, that is

$$h(i) = |u_i/v_i|$$

where

$$u_i = x_i\left[y_{i-(w-1/2)} - y_{i+(w-1/2)}\right]$$

$$+y_i\left[x_{i+(w-1/2)} - x_{i-(w-1/2)}\right]$$

$$+\left[y_{i+(w-1/2)}\right]\left[x_{i+(w-1/2)}\right]$$

$$-\left[y_{i-(w-1/2)}\right]\left[x_{i+(w-1/2)}\right],$$

and

$$v_i = [\left(y_{i-(w-1/2)} - y_{i+(w-1/2)}\right)^2$$

$$+\left(x_{i-(w-1/2)} - x_{i+(w-1/2)}\right)^2]^{1/2}$$

Compution of local contour sequence $h(i)$ for N points is shown in figure (3.4) .we computed $h(i)$ for each i and aray of $h(i)$ is represented by H(i) , if there is N pixeal in contour then, [1]

$$\mathrm{H}(i) = [h(1), h(2) \dots \dots \dots . h(N)]$$

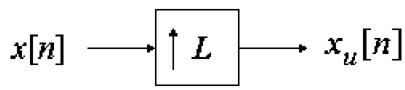
3.4 NORMALIZATION OF LOCAL COUNTER SEQUENCE

During creating database there is no restricions are placed on the position , distance , and orientation of gesture in front of camera. LSC is invarient to translation *i.e.* if there is change in position of a gesture. Start-point is determined by locating the first contour pixel using a left-to-right and top-to-bottom scan of the image. Therefore, a change in the orientation of a gesture results in a circular shift in the samples. The no of pixel in contour varrys according thr distance of the gesture from camera thus the scaling of the amplitude of the LCS can be easily normalized by dividing the samples of the LCS by the standard deviation of the LCS[1].

The scaling of the duration can also be normalized by uniformaly expanding or compressing the LCS to have a fixed duration $\widehat{N}$ using a up or down samplar [1], [4].

3.4.1 UP SAMPLER

An up-sampler with an up-sampling factor L, where L is a positive integer, develops an output sequenc $x_u(n)$ with a sampling rate that is L times larger than that of the input sequence $x(n)$ [4].Up-sampling operation is implemented by inserting *(L-1)*equidistant zero-valued samples between two consecutive samples of $x(n)$

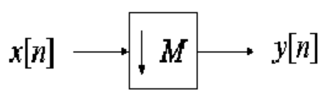


Input optput relation of an up sampler is given by

$$x_u(n) = \begin{cases} x[n/L], & andn = 0, \pm L \pm 2L, \ldots \\ 0, & andothervise \end{cases}$$

3.4.2 DOWN SAMPLER

An down-sampler with a down-sampling factor M, where M is a positive integer, develops an output sequence $y[n]$ with a sampling rate that is *(1/M)*th of that of the input sequence *x[n]*. Down-sampling operation is implemented by keeping every M-th sample of x[n] and removing *(M-1)* in-between samples to generate $y(n)$



Input-output relation of a down sampler is given by

$$y[n] = x[nM]$$

3.5 ADVANTAGES OF LOCAL CONTOUR SEQUENE

1. The local contour sequence (LCS) is computed in a sucha way that it does not depend on shape copmplexity so its sutainle for gesture which is having convex and concave contour

2. Some times hand is not parallel to camera so some part of gesture is obscured but LCS can also be used to robustly represent partial contour [5] .So due to this characteristic of LCS the visible part of gesture will not be affected.

3. LCS representation does not have any derivative comptution as slopes [6] or curvature,the it is roboust with respect contour noise (random variations in thecontour).

4. Increasing w tends to increase the amplitudes of the samples of the localized contour sequence. An increase in the amplitudes has the effect of increasing the signal-to-noise ratio for a fixed contour noise level, therefore, the robustness with respect to contour noise can be increased by increasing *w* [1].

3.6 RESULTS and SIMULATION

3.6.1 CONTOUR DETECTION RESULT

After removing the noises in the segmented image we applied canny edge detection algorithm to find the contour of the image and then a contour tracking algorithm is applied to give the pixel of the boundary to a sequential order.

Fig 3.5 Contour of gesture "a"

Fig 3.6 Contour of gesture "b"

Fig 3.7 Contour of gesture “c”

Fig 3.8 Contour of gesture “d”

3.6.2 LOCAL CONTOUR SEQEUNCE RESULT

A contour tracking algorithm is applied [1] to track the contour and for numbering contour pixel in sequential order. The LCS is for every pixel is computed by using eq. in chapter 3.Then LCS were normalized by dividing the samples of the LCS by the standard deviation of the LCS. Duration is normalized by using linear transformation

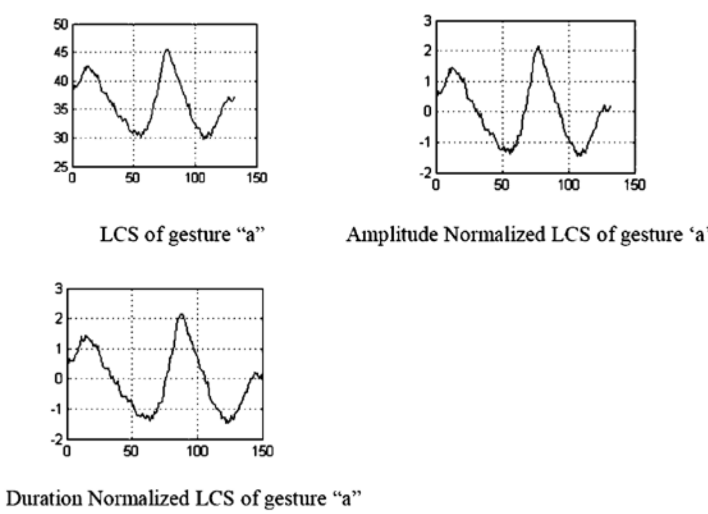
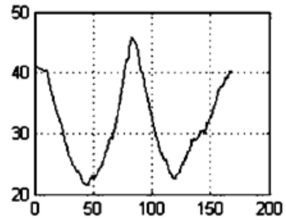
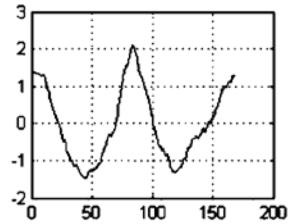


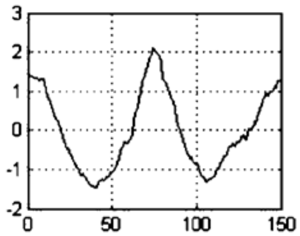
Fig3.9LCS of gesture “a”



LCS of gesture "b"

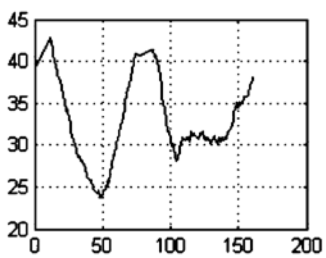


Amplitude Normalized LCS of gesture "b"

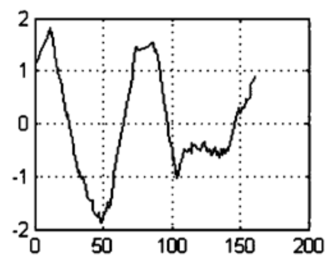


Duration Normalized LCS of gesture "b"

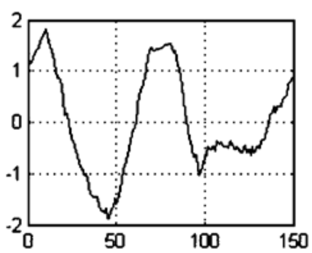
Fig 3.10 LCS of gesture "b"



LCS of gesture "c"

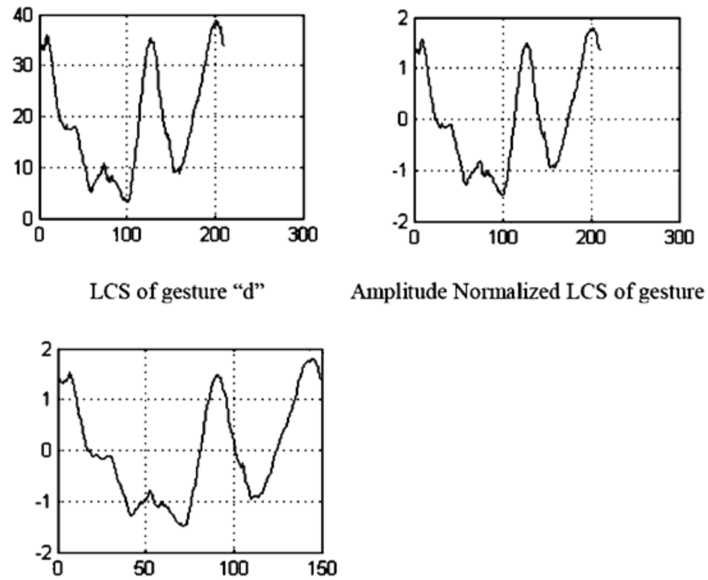


Amplitude Normalized LCS of gesture "c"



Duration Normalized LCS of gesture "c"

Fig 3.11 LCS of Gesture "c"



LCS of gesture “d”

Amplitude Normalized LCS of gesture “d”

Duration Normalized LCS of gesture “d”

Fig 3.12 LCS of Gesture “d”

3.7 CONCLUSION

Feature extraction is very important step in gesture recognition system. In this chapter feature extraction method is described. We have used Local contour sequence as our prime feature. Canny edge detection technique is used to detect the border of hand in image. A contour tracking is applied to find the contour and pixel in contour is numbered sequentially. Local contour sequence for any arbitrary pixel is calculated as perpendicular distance from the chord connecting end points of window size *w*.

REFERENCES

[1] Lalit Gupta and Suwei Ma, “Gesture-Based Interaction and Communication: Automated Classification of Hand Gesture Contours,” *IEEE transactions on systems, man, and cybernetics—part c: applications and reviews*, vol. 31, no. 1, February 2001

[2] J. Canny, “A computational approach to edge detection,” *IEEE Trans. Pattern Anal. Machine. Intell*., vol. 8, no. 6, pp. 679–698, Nov. 1986.

[3] Bill Green “Canny Edge Detection Tutorial.” , http://www.pages.drexel.edu/~weg22/can_tut.html ,2002

[4] R. K. Cope and P. I. Rockett, “Efficacy of gaussian smoothing in Canny edge detector,” Electron. Lett., vol. 36, pp. 1615–1616, 2000

[5] L. Gupta, T. Sortrakul, A. Charles, and P. Kisatsky, “Robust automatic target recognition using a localized boundary representation,” Pattern Recognit., vol. 28-10, pp. 1587–1598, 1995

[6] Y. Yitzhaky and E. Peli, “A method for objective edge detection evaluation and detector parameter selection,” *IEEE Trans. Pattern Anal. Machine Intell*., vol. 25, no. 8, pp. 1027–1033, Aug. 2003.

[7] P. I. Rockett, “Performance assessment of feature detection algorithms: Amethodology and case study on corner detectors,” *IEEE Trans. Image Process*., vol. 12, no. 11, pp. 1668–1676, Nov. 2003.

[8] D. J. Sturman and D. Zeltzer, “A survey of glove-based input, gesture recognition system” *IEEE Comput. Graph. Appl.*, vol. 14, pp. 30–39, Jan. 1994.

[9] L. Gupta and K. Malakapalli, “Robust partial shape classification using invariant breakpoints and dynamic alignment,” Pattern Recognit., vol. 23-10, pp. 1103–1111, 1990.

[10] H. Sakoe and S. Chiba, “Dynamic programming optimization for spoken word recognition,” *IEEE Trans. Acoust. Speech Signal Processing*, vol. ASSP-26, pp. 43–49, Feb. 1978.

CHAPTER 4

CLASSIFICATION

4.1 LINEAR CLASSIFIER

In this project we have applied linear alignment technique for classification of different hand gesture .In this method we computed the similarity between local contour sequence(LCS).we have already normalized the LCS by using the standard deviation and linear transformation . So LCS for all gesture is amplitude and duration normalized [1].Let us suppose Total class is represented by M $(i = 1.2 \ldots \ldots M)$, then m is a fully amplitude and duration normalized LCS of a reference gesture of a class m and a test gesture is represented by $\hat{h}_m(i)$ and $\hat{t}(i), i = 1,2 \ldots \ldots \widehat{N}$, respectivily , then ,the dissimililarity between the two LCSs is obtained by first determining[1]

$$D_m(j) = \sum_{i=1}^{\widehat{N}} \left| \hat{h}_m(i) - \hat{t}\big((i + j)\big) \right|$$

Where $j = 0,1, \ldots\ldots (\widehat{N} - 1)$

Here $\hat{t}\big((i + j)\big)$ denotes a circular shift of j samples in $\hat{t}(i)$. $D_m(j)$ is computed between refrence gesture and test gesture The best match between $\hat{h}_m(i)$ and $\hat{t}(i)$ is then given by

$$D_m = \min_j D_m(j)$$

The test gesture is tends to belong to each gesture class to compute $D_m, m = 1,2, \ldots . M$; and the test gesture is assigned to class m^* is given by the minimum distance rule

$$m^* = \arg\min D_m$$

4.2 SUPPORT VECTOR MACHINE

Machine learning is known as subfield of artificial intelligence. Through machine learning we can develop methods for enabling a computer to learn. Over the period there are so many techniques developed for machine learning.

Support vector machine (SVM) has been firstly introduced by Vapnik [1] and gained popularity because of its exiting feature such as better empirical performance. Support vector machine (SVM) is a classification and regression technique that uses machine learning theory to maximize the accuracy of prediction [2].

In this chapter we discuss support vector machines for two-class problems. First, we discuss support vector machines, in which training data are linearly separable in the input space [3].

Then we discuss support vector machines for the case where training data are not linearly separable and map the input space into the high-dimensional feature space to enhance linear separability in the feature space [2]. For a two-class problem, a support vector machine is trained so that the direct decision function maximizes the generalization ability namely, the m -dimensional input space x is mapped into the l dimensional l ≥ m feature space z. Then in z , the quadratic programming problem is solved to separate two classes by the optimal separating hyperplane [2].

Let M m -dimensional training inputs $x_i(i = 1, \dots \dots . M)$ belong to Class 1 or 2 and the associated labels are $y_i = 1$ for Class 1 and -1 for Class 2. If these data are linearly separable, we can determine the decision function [1].

$$D(x) = w^T x + b, \tag{4.2.1}$$

Where w is an m -dimensional vector, b is a bias term, and for $i = 1, \dots \dots . M$

If training data is linearly separable, no training data satisfy $w^T x + b = 0$ .Thus we consider the following inequalities

$$w^T x_i + b \begin{cases} \geq 1 & for \quad y_i = 1 \\ \leq 1 & for \quad y_i = -1 \end{cases} \tag{4.2.2}$$

We can write eq. in generalize form

$$y_i(w^T x_i + b) \geq 1 \qquad for\ i = 1, \dots \dots . M \tag{4.2.3}$$

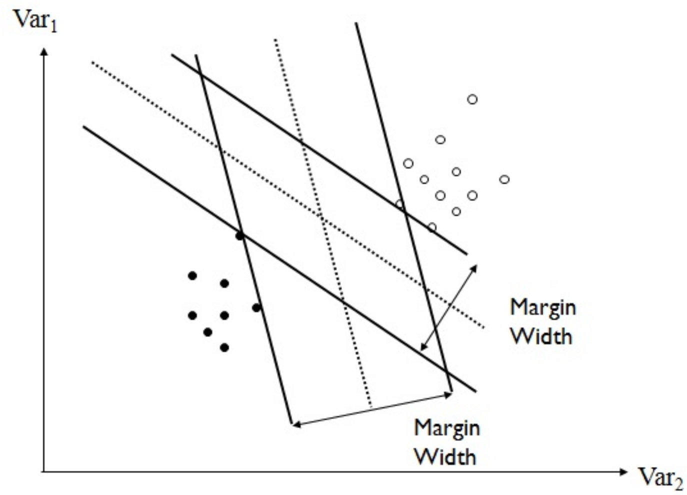


Fig 4.1 linear SVM representation

Figure 2.1 shows two decision functions that satisfy (4.2.2). Thus there are an infinite number of decision functions that satisfy (4.2.3), which are separating hyperplanes. The generalization ability depends on the location of the separating hyperplane, and the hyperplane with the maximum margin is called the optimal separating hyperplane [1]. Therefore, the optimal separating hyperplane can be obtained by solving the following minimization problem for w and b:

$$minimize \quad Q(w, b) = \frac{1}{2}\|w\|^2 \tag{4.2.4}$$

$$\text{subjected} \quad y_i(w^T x_i + b) \geq 1 \quad for \text{ i} = 1, \dots \dots \text{M} \tag{4.2.5}$$

Here, the square of the Euclidean norm ||w|| in (4.2.4) is to make the optimization problem quadratic programming. The assumption [4] of linear separability means that there exist w and b that satisfy (4.2.5) known as feasible solutions. To do this, we first convert the constrained problem given by (4.2.4) and (4.2.5) into the quadratic problem:

$$Q(w, b, \alpha) = \frac{1}{2} w^T w - \sum_{i=1,}^{M} \alpha_i \{y_i(w^T x_i + b) - 1\}, \tag{4.2.6}$$

Where $\alpha = (\alpha_i \dots\dots \alpha_M)^T$and α_i are the nonnegative Lagrange multipliers [3]. The optimal solution of is given by the saddle point, where (4.2.4) is minimized with respect to w, maximized with respect to $\alpha_i (\geq 0)$, and maximized or minimized with respect to b according to the sign of $\sum_i^M \alpha_i y_i$ and the solution satisfies the following Karush–Kuhn–Tucker (KKT) conditions [1]

$$\frac{\partial Q(w,b,\alpha)}{\partial w} = 0, \tag{4.2.7}$$

$$\frac{\partial Q(w,b,\alpha)}{\partial b} = 0, \tag{4.2.8}$$

$$\alpha_i \{y_i(w^T x_i + b) - 1\} = 0 \quad for\ i = 1, \dots\dots M, \tag{4.2.9}$$

$$\alpha_i \geq 0 \ \ for\ i = 1, \dots\dots M \tag{4.2.10}$$

From (2.14), $\alpha_i = 0$, or $\alpha_i \neq 0$,and $y_i(w^T x_i + b)$ must be satisfied. The training data x_i with $\alpha_i = 0$ are called support vectors. Using (4.2.7), we reduce (4.211) and (4.2.12), respectively, to

$$w = \sum_{i=1}^{M} \alpha_i y_i\, x_i \tag{4.2.11}$$

And

$$\sum_{i=1}^{M} \alpha_i y_i = 0 \tag{4.2.12}$$

Substituting (4.2.11) and (4.2.12) into (4.2.7), we obtain the following dual problem:

$$maximize \quad Q(\alpha) = \sum_{i=1}^{M} \alpha_i - \frac{1}{2} \sum_{j=1}^{M} \alpha_i \alpha_j\, y_i y_j x_i^T x_j \tag{4.2.8}$$

$$subjected\ to \quad \sum_{i=1}^{M} y_i\, \alpha_i = 0, \quad \alpha_i \geq 0 \ \ for\ i = 1, \dots\dots\dots M \tag{4.2.9}$$

The formulated support vector machine is called the hard-margin support vector machine. Maximizing (4.2.8) under the constraints (4.2.9) is a concave quadric programming problem [5] we will find, the global optimum solutions $\alpha_i(i-1,\ldots\ldots M)$, now our decision function becomes

$$D(x) = \sum_{i\epsilon s} \alpha_i y_i x_i^T x + b, \tag{4.2.10}$$

Where S is the set of support vector indices, and from the KKT conditions given by (2.2.9), b is given by

$$b = y_i - w^T x_i \qquad for\ i\ \epsilon\ S \tag{4.2.11}$$

For calculation we take average [1]

$$b = \frac{1}{|S|} \sum_{i\epsilon s}(y_i - w^T x_i) \tag{4.2.12}$$

Then unknown data sample X is classified into

$$\begin{cases} class\ 1 & if \quad D(X) > 0, \\ class - 1 & if \quad D(X) < 0. \end{cases} \tag{4.2.13}$$

In real world problem it is not likely to get an exactly separate line dividing the data within the space. And there might have a curved decision boundary [5]. We might have a hyperplane which might exactly separate the data but this may not be desirable if the data has noise in it. It is better for the smooth boundary to ignore few data points than be curved or go in loops, around the outliers [1]. To allow inseparability, we introduce the nonnegative slack variables $\xi_i(\geq 0)$ into (4.2.3):

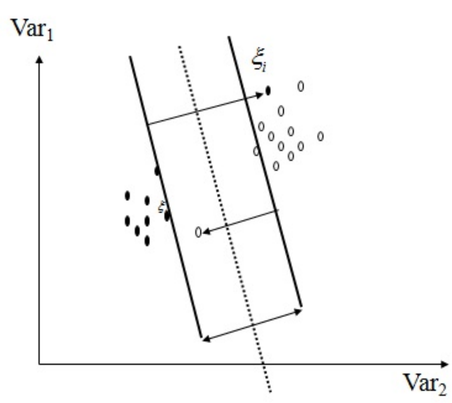


Fig4.2 nonseperable SVM representation

$$y_i(w^T x_i + b) \geq 1 - \xi_i \qquad for\ \mathrm{i} = 1, \ldots \ldots \mathrm{M} \tag{4.2.14}$$

To obtain the optimal hyperplane in which the number of training data that do not have the maximum margin is minimum, we need to minimize [4]

$$Q(w) = \sum_{i=1}^{M} \theta(\xi_i)$$

Where

$$\theta(\xi_i) = \begin{cases} 1 & for \quad \xi_i > 0, \\ 0 & for \quad \xi_i = 0. \end{cases}$$

We consider the following minimization problem:

$$minimize \quad Q(w, b, \xi) = \frac{1}{2}\|w\|^2 + \frac{C}{p}\sum_{i=1}^{M} \xi_i^p \tag{4.2.15}$$

$$subjected \quad y_i(w^T x_i + b) \geq 1 - \xi_i \quad for \text{ i} = 1, \dots\dots \text{M} \tag{4.2.16}$$

Where $\xi = \left((\xi_i \dots \dots \xi_M)\right)^T$ and C is the margin parameter that determines the trade-off between the maximization of the margin and the minimization of the classification error. We select the value of p as either 1 or 2. We call the obtained hyperplane the soft-margin hyperplane [1]. When p = 1, we call the support vector machine the L1 soft-margin support vector machine or the L1 support vector machine for short (L1 SVM) and when p = 2, the L2 soft-margin support vector machine or L2 support vector machine (L2 SVM) [1]. Similar to the linearly separable case, introducing the nonnegative Lagrange multipliers α_i and β_i, we obtain eq (4.2.17)

$$Q(w, b, \xi, \alpha, \beta) = C\sum_{i=1}^{M} \xi_i$$
$$- \sum_{i=1}^{M} \alpha_i(y_i(w^T x_i + b) - 1 + \xi_i) - \sum_{i=1}^{M} \beta_i \xi_i \tag{4.2.17}$$

Where $\alpha = (\alpha_i \dots\dots \alpha_M)^T$ and $\beta = (\beta_i \dots\dots \beta_M)^T$

For optimal solution, following KKT condition are satisfied

$$\frac{\partial Q(w,b,\xi,\alpha,\beta)}{\partial w} = 0 \tag{4.2.18}$$

$$\frac{\partial Q(w,b,\xi,\alpha,\beta)}{\partial b} = 0 \tag{4.2.19}$$

$$\frac{\partial Q(w,b,\xi,\alpha,\beta)}{\partial \xi} = 0 \tag{4.2.20}$$

$$\alpha_i\{y_i(w^T x_i + b) - 1 + \xi\} = 0 \quad for\ i = 1, \dots\dots M, \tag{4.2.21}$$

$$\alpha_i \xi_i = 0 \quad for\ i = 1, \dots\dots M, \tag{4.2.22}$$

$$\alpha_i \geq 0, \;\; \beta_i \geq 0, \;\; \xi_i \geq 0 \quad for\ i = 1, \dots\dots M. \tag{4.2.23}$$

Weight matrix can be computed as

$$w = \sum_{i=1}^{M} \alpha_i y_i x_i \tag{4.2.24}$$

And

$$\sum_{i=1}^{M} \alpha_i y_i = 0 \tag{4.2.25}$$

$$\alpha_i + \beta_i = 0 \quad for\ i = 1,2 \ldots \ldots ..M \tag{4.2.26}$$

Thus substituting (4.2.24), (4.2.25), and (4.2.26) into (4.2.15), we obtain the following dual problem:

$$maximize \quad Q(\alpha) = \sum_{i=1}^{M} \alpha_i - \frac{1}{2}\sum_{i,j=1}^{M} \alpha_i \alpha_j\, y_i y_j x_i^T x_j \tag{4.2.27}$$

$$subjected\ to \quad \sum_{i=1}^{M} y_i\, \alpha_i = 0, \quad C \geq\ \alpha_i \geq 0 \quad for\ i = 1, \ldots \ldots ..M \tag{4.2.28}$$

The only difference between *L1* soft-margin support vector machines and hard-margin support vector machines is that α_i cannot exceed C. The inequality constraints in *(4.2.21)* are called box constraints. Especially, (4.2.22*)* and (4.2.23*)* are called *KKT* (complementarily) conditions.

From these and (4.2.25*)*, there are three cases for α_i[1]:

Case1: $\alpha_i = 0$. Then $\xi_i = 0$. Thus x_i is correctly classified.

Case2: $0 < \alpha_i < C$ Then $y_i(w^T x_i + b) - 1 + \xi = 0$ and $\xi_i = 0$. Therefore, $y_i(w^T x_i + b) = 1$ and x_i is a support vector. Especially, we call the support vector with $0 > \alpha_i > C$ an unbounded support vector.

Case3*:* $\alpha_i = C$ Then $y_i(w^T x_i + b) - 1 + \xi = 0$ and ≥ 1 . Thus x_i is a support vector. We call the support vector with: $\alpha_i = C$ a bounded support vector. If, : $0 \leq \xi_i < 1$, x_i is correctly classified, and if$\xi_i \geq 1$, x_i is misclassified.

The decision function is the same as that of the hard-margin support vector machine and is given by

$$D(x) = \sum_{i \epsilon S} \alpha_i y_i x_i^T x + b \tag{4.2.29}$$

Where S is the set of support vector indices. Because α_i are nonzero for the support vectors, the summation in (2.53) is added only for the support vectors. For the unbounded α_i,

$$b = y_i - w^T x_i \tag{4.2.30}$$

By taking average of b

$$b = \frac{1}{|U|}\sum_{i \epsilon U}(y_i - w^T x_i) \tag{4.2.31}$$

Where U is unbounded support vector. The unknown data sample x is classified into

$$\begin{cases} class\ 1 & if \quad D(X) > 0, \\ class\ 1 & if \quad D(X) < 0. \end{cases} \tag{4.2.32}$$

If data is linear, a separating hyper plane may be used to divide the data. However it is often the case that the data is far from linear and the datasets are inseparable. To allow for this kernels are used to non-linearly map the input data to a high-dimensional space. The new mapping is then linearly separable [7]

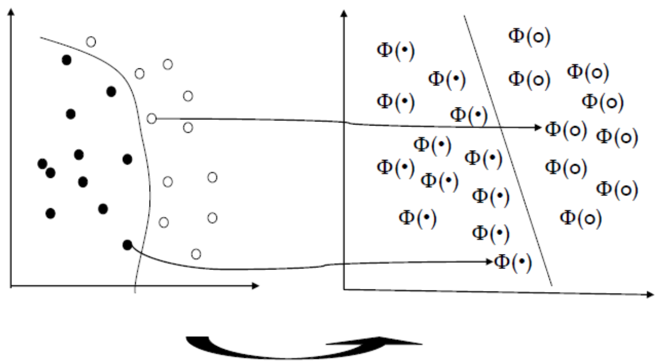


Fig 4.3 transform from input space to feature space

using the nonlinear vector function $\phi(\mathrm{x}) = \left(\phi_1(\mathrm{x}), \ldots \ldots \phi_l(\mathrm{x})\right)^{\mathrm{T}}$ that maps the m-dimensional input vector x into the l-dimensional feature space, the linear decision function in the feature space is given by

$$\mathrm{D}(\mathrm{x}) = \mathrm{w}^{\mathrm{T}}\phi(\mathrm{x}) + \mathrm{b}, \tag{4.2.33}$$

Where *w* is an l-dimensional vector and b is a bias term.

$$K(x, x') = \phi^T(x)\phi(\mathrm{x}'). \tag{4.2.34}$$

Here $K(x, x')$ is a mapping function. The advantage of using kernels is that we need not treat the high dimensional feature space explicitly. This technique is called kernel trick, namely, we use $K(x, x')$ in training and classification instead of $\phi(x)$. The methods that map the input space into the feature space and avoid explicit treatment of variables in the feature space by kernel tricks are called kernel methods or kernel-based methods. Using the kernel, the dual problem in the feature space is given as follows [1]:

$$maximize \quad Q(\alpha) = \sum_{i=1}^{M} \alpha_i - \frac{1}{2}\sum_{i,j=1}^{M} \alpha_i\alpha_j\, y_i y_j \mathrm{K}(\mathrm{x}, \mathrm{x}') \tag{4.2.35}$$

$$subjected\ to \quad \sum_{i=1}^{M} y_i\, \alpha_i = 0, \quad C \geq \alpha_i \geq 0 \quad for\ i = 1, \ldots \ldots .. M \tag{4.2.36}$$

For optimal solution, following KKT condition are satisfied

$$\alpha_i\{y_i(\sum_{j=1}^{M} y_j\alpha_j \,\mathrm{K}(\mathrm{x},\mathrm{x}') + b) - 1 + \xi\} = 0 \quad for\ i = 1, \dots\dots M, \tag{4.2.37}$$

$$(C - \alpha_i)\xi_i = 0 \quad for\ i = 1, \dots\dots M, \tag{4.2.38}$$

$$\alpha_i \geq 0,\ \ \xi_i \geq 0 \quad for\ i = 1, \dots\dots M. \tag{4.2.39}$$

The decision function is given by

$$D(x) = \sum_{i\epsilon s} \alpha_i y_i K(x, x') + b \tag{4.2.40}$$

Where b is given by

$$\mathrm{b} = \mathrm{y_i} - \sum_{\mathrm{i}\epsilon \mathrm{s}} \alpha_\mathrm{i} \mathrm{y_i} \mathrm{K}(\mathrm{x}, \mathrm{x}') \tag{4.2.41}$$

To ensure stability of calculations, we take the average:

$$b = \frac{1}{|U|} \sum_{i\epsilon U} (y_i \sum_{i\epsilon U} \alpha_i y_i K(x, x')) \tag{4.2.42}$$

Where U is unbounded support vector. The unknown data sample x is classified into

$$\begin{cases} class\ 1 & if \quad D(X) > 0, \\ class\ 1 & if \quad D(X) < 0. \end{cases} \tag{4.2.43}$$

Kernels

One of benefits of SVM is that we can generalize different types of problem by taking different type of kernel. Some kernel descriptions are given below. In this project we have used Gaussian

1. Linear kernel [3]:

Linear kernel is for linearly separable case where we do not have to map the input space to a feature space which is high- dimensional, linear kernel has following form

$$K(x, x') = x^T x'$$

2 Polynomial kernels*:*

Polynomial kernel is used for non-linear modeling .It is used because it avoids problem of having hessian. Polynomial kernel has following form

$$K(x, x') = (x^T x' + 1)^d$$

3 Exponential Radial Basis Function:

Radial basis functions most commonly with a Gaussian form [6]

$$K(x, x') = exp\left(-\frac{\|x - x'\|}{2\sigma^2}\right)$$

4 Multi-layers Perceptron:

The long established MLP, with a single hidden layer, also has a valid kernel representation [6].

$$K(x, x') = tanh(\rho(x, x') + e)$$

4.3 MULTICLASS SUPPORT VECTOR MACHINES

Support Vector Machines (SVM) learning machine based on the structural risk minimization induction principle[2] The conventional way to extend it to multi-class scenario is to decompose an *M*-class problem into a series of two-class problems, for which one-against-all is the earliest and one of the most widely used implementations[2]. One drawback of this method, however, is that when the results from the multiple classifiers [8] are combined for the final decision

Let the i^{th} decision function, with the maximum margin that separates class i from the remaining classes, be

$$D_i(x) = w_i^T \phi(x) + b_i$$

Where w_i is the l-dimensional vector, $\phi(x)$ is the mapping function that maps x into the l - dimensional feature space, andi is the b_i bias term [8]. The hyperplane forms the optimal separating hyperplane, and if the classification problem is separable, the training data belonging to class i satisfy $D_i(x) = 0$ and those belonging to the remaining classes satisfy$D_i(x) \leq 1$. If the problem is inseparable, unbounded support vectors satisfy $|D_i(x)| = 1$ and bounded support vectors belonging to class i satisfy $D_i(x) \leq 1$ and those belonging to a class other than class i $D_i(x) \geq -1$. Data sample *x* is classified into the class

$$\arg \max_{i=1,\ldots.n} D_i(x).$$

4.4 LEAST-SQUARE SUPPORT VECTOR MACHINES

As the expansion of the standard Support Vector Machine, compared with the traditional standard Support Vector Machine, the Least Squares Support Vector Machine [9] loses the sparseness of standard Support Vector Machine, which would affect the in increased efficiency [3]. For a two-class problem, we consider the following decision function [2]:

$$D(x) = w^T \phi(x) + b \qquad (4.4.1)$$

Where w is the l-dimensional vector, b is the bias term, and $\phi(x)$ is the l dimensional vector that maps m-dimensional vector x into the feature space. If $D(x) > 0$, x is classified into Class 1 and if $D(x) < 0$, Class 2. The LS support vector machine is formulated as follows [1]:

$$minimize \qquad \frac{1}{2}w^T w + \frac{C}{2}\sum_{i=1}^{M}\xi_i^2 \tag{4.4.2}$$

$$subjected\ to \quad y_i(w^T\phi(x_i) + b) = 1 - \xi_i^2 \quad for\ i = 1, \ldots\ldots M, \tag{4.4.3}$$

Where (x_i, y_i) $(i = 1, \ldots\ldots M)$, M are training input–output pairs, $y_i = 1$ or -1 if xi belongs to Class 1 or 2, respectively, ξ_i are the slack variables for x_i , and C is the margin parameter. Multiplying y_i to both sides of the equation in (4.3.3), we obtain

$$y_i - w^T\phi(x_i) - b = y_i\xi_i \tag{4.4.4}$$

Because ξ_i takes either a positive or a negative value and $|y_i| = 1$, , instead of (4.4.3), we can use

$$y_i - w^T\phi(x_i) - b = \xi_i \tag{4.4.5}$$

Introducing the Lagrange multipliers α_i into (4.3.2) and (4.3.5), we obtain the unconstrained objective function [2]:

$$Q(w, b, \alpha, \xi) = \frac{1}{2}w^T + \frac{C}{2}\sum_{i=1}^{M}\xi_i^2 - \sum_{i=1}^{M}\alpha_i(w^T\phi(x_i) + b - y_i + \xi_i), \tag{4.4.6}$$

Where $\alpha = (\alpha_1, \ldots \ldots \alpha_M)^T$ and $\xi = (\xi_1, \ldots \ldots \xi_M)^T$. Taking the partial derivatives of (4.3.6) with respect tow , b and ξ and equating them to zero [10], together with the equality constraint (4.3.5), we obtain the optimal conditions as follows [2]:

$$w = \sum_{i=1}^{M}\alpha_i\phi\,(x_i), \tag{4.4.7}$$

$$\sum_{i=1}^{M}\alpha_i = 0, \tag{4.4.8}$$

$$\alpha_i = C\xi_i \quad for\ i = 1, \ldots M, \tag{4.4.9}$$

$$w^T\phi(x_i) + b - y_i + \xi_i = 0 \qquad for\ i = 1, \ldots\ldots\ldots M \tag{4.4.10}$$

α_i can be negative in LSSVM [1]. Substituting (4.3.7) and (4.3.9) into (4.3.10) and expressing it and (4.3.8) in matrix form, we obtain eq.

$$\begin{pmatrix} \Omega & 1 \\ 1 & 0 \end{pmatrix}\begin{pmatrix} \alpha \\ b \end{pmatrix} = \begin{pmatrix} y \\ 0 \end{pmatrix} \tag{4.4.11}$$

Or

$$\Omega\alpha + 1b = y, \tag{4.4.12}$$

$$1^T\alpha = 0, \tag{4.4.13}$$

Where 1 is the M-dimensionalvector and

$$\Omega_{ij} = \phi^T(x_i)\phi(x_j) + \frac{\delta_{ij}}{C}, \tag{4.4.14}$$

$$\delta_{ij} = \begin{cases} 1 & i = j, \\ 0 & i \neq j, \end{cases} \tag{4.4.15}$$

$$y = (y_1, \ldots\ldots\ldots y_M)^T \tag{4.4.17}$$

Here α value can be deducted from eq. (4.3.12)

$$\alpha = \Omega^{-1}(y - 1b) \tag{4.4.18}$$

Substituting (4.4.11) into (4.4.12), we obtain

$$b = (1^T \Omega^{-1} 1)^{-1} 1^T \Omega^{-1} y \tag{4.3.19}$$

Decision function for LSSVM is given as

$$D(x) = \alpha^T \varphi(x) + b = \sum_{1=1}^{M} \alpha_i \, K(x, x_i) + b, \tag{4.3.20}$$

4.5 RESULT

After finding the LCS we fed all images LCS to the different classifier .classification is done in two phases first is training phase and second one is testing. we have total database of 1000 hand gesture images constructing 25 class so each class has 40 images .we used 20 images per class For train the classifier and 20 images per class for test the images

4.5.1 CLASSFICATION RESULT USING LINEAR CLASSIFIER

Dissimilarity between the two *LCSs* is obtained by first determining

$$D_m(j) = \sum_{i=1}^{\hat{N}} \left| \hat{h}_m(i) - \hat{t}\big((i + j)\big) \right|$$

Here $D_m(j)$ represent dissimilarity between reference gesture and test gesture we computed taking every gesture as a reference and test all the gesture with reference gesture. The and we computed $D_m(j)$ for every gesture

The best match between $\hat{h}_m(i)$ and$\hat{t}(i)$ is then given by

$$D_m = \min_j D_m(j)$$

The test gesture is tends to belong to each gesture class to compute $\mathrm{D_m}, \mathrm{m} = 1,2, \ldots . \mathrm{M};$ and the test gesture is assigned to class m^* given by the minimum distance rule

$$m^* = \arg\min D_m$$

In confusion table total 500 gesture were tested (20 each gesture).Confusion matrix is given below

	TEST GESTURE																								
	a	ae	b	c	d	e	f	g	h	i	k	l	m	n	o	p	q	r	s	t	u	v	w	x	y
a	20	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
ae	0	18	0	0	1	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
b	0	1	17	0	0	0	0	0	0	0	1	0	0	0	1	0	0	0	0	0	0	0	0	0	0
c	0	0	0	20	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
d	0	0	0	0	18	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0
e	0	0	1	0	0	18	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0
f	0	0	0	0	0	0	20	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
g	0	0	0	0	0	0	0	19	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0
h	0	0	0	0	0	0	0	0	19	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
i	0	0	0	0	0	0	0	0	0	17	0	1	0	0	1	0	0	1	0	0	0	0	0	0	0
k	0	0	0	0	0	0	0	0	0	0	20	0	0	0	0	0	0	0	0	0	0	0	0	0	0
l	0	0	0	0	0	0	0	0	1	0	0	19	0	0	0	0	0	0	0	0	0	0	0	0	0
m	0	0	0	0	0	0	0	0	0	0	0	0	19	0	0	0	0	1	0	0	0	0	0	0	0
n	0	0	0	0	0	0	0	0	0	1	0	0	0	18	1	0	0	0	0	0	0	0	0	0	0
o	0	0	0	0	0	0	0	0	0	0	0	0	0	0	20	0	0	0	0	0	0	0	0	0	0
p	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	20	0	0	0	0	0	0	0	0	0
q	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	18	1	0	0	0	0	0	0	0
r	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	20	0	0	0	0	0	0	0
s	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	20	0	0	0	0	0	0
t	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	18	1	0	0	0	0
u	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	20	0	0	0	0
v	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	20	0	0	0
w	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	18	1	0
x	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	19	0
y	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	1	0	0	0	18

Table (4.1) confusion matrix of linear classifier

$$\text{Accuracy}=\frac{gesture\ classfied\ correctly}{total\ gesture}\times 100\%$$

Gesture classified correctly=473

Total gesture=500

$$\text{Accuracy}=\frac{473}{500}\times 100 = 94.6\%$$

4.5.2 CLASSIFICATION RESULT USING MULTI CLASS SUPPORT VECTOR MACHINE

While training the SVM we fed 20 images' LCS foe train the SVM.We used MATLAB for our simulation. After training we fed LCS of test images one by one and using multiclass SVM algorithm we classify class for each image Program is written in MATLAB using SVM toolbox [11]. This is done by multiclass SVM algorithm which we have discussed in chapter 4.Here total class is 25 .Confusion matrix of SVM is found as below:

	TEST GESTURE																								
	a	ae	b	c	d	e	f	g	h	i	k	l	m	n	o	p	q	r	s	t	u	v	w	x	y
a	20	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
æ	0	20	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
b	0	0	20	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
c	0	0	0	19	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0
d	0	0	0	0	18	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0
e	0	0	1	0	0	18	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0
f	0	0	0	0	0	0	20	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
g	0	0	0	0	0	0	0	19	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0
h	0	0	0	0	0	0	0	0	20	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
i	0	0	0	0	0	0	0	0	0	20	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
k	0	0	0	0	0	0	0	0	0	0	20	0	0	0	0	0	0	0	0	0	0	0	0	0	0
l	0	0	0	0	0	0	0	0	1	0	0	20	0	0	0	0	0	0	0	0	0	0	0	0	0
m	0	0	0	0	0	0	0	0	0	0	0	0	20	0	0	0	0	0	0	0	0	0	0	0	0
n	0	0	0	0	0	0	0	0	0	0	0	0	0	20	0	0	0	0	0	0	0	0	0	0	0
o	0	0	0	0	0	0	0	0	0	0	0	0	0	0	20	0	0	0	0	0	0	0	0	0	0
p	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	20	0	0	0	0	0	0	0	0	0
q	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	20	0	0	0	0	0	0	0	0
r	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	20	0	0	0	0	0	0	0
s	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	20	0	0	0	0	0	0
t	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	20	0	0	0	0	0
u	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	20	0	0	0	0
v	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	20	0	0	0
w	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	20	0	0
x	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	19	0
y	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	20

Table (4.2) Confusion matrix of Multiclass Support Vector Machine

Number of gesture per class=20

Total class=25

Total no. gesture=20*25=500

Correctly classified gesture =493

Misclassified gesture=7

Accuracy = $(\frac{correctly\ classfied\ gesture}{total\ no.of\ gesture}) \times 100\%$

$$= \left(\frac{493}{500}\right) \times 100 = 98.6\%$$

4.5.3 CLASSIFICATION RESULT USING MULTICLASS LEAST SQUARE SUPPORT VECTOR MACHINE

We hav given the same training gesture for LSSVM and same test data .We hav used LSSVM [12] toolbox for the classificatin.confusion matrix is given below.

	TEST GESTURE																							
	a	ae	b	c	d	e	f	g	h	i	k	l	m	n	o	p	q	r	s	t	u	v	w	x	y
a	20	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
æ	0	19	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0
b	0	0	20	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
c	0	0	0	19	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
d	0	0	0	0	20	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
e	0	0	0	0	0	20	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
f	0	0	0	0	0	0	20	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
g	0	0	0	0	0	0	0	20	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
h	0	0	0	1	0	0	0	0	19	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
i	0	0	0	0	0	0	0	0	0	20	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
k	0	0	0	0	0	0	0	0	0	0	20	0	0	0	0	0	0	0	0	0	0	0	0	0	0
l	0	0	0	0	0	0	0	0	1	0	0	20	0	0	0	0	0	0	0	0	0	0	0	0	0
m	0	0	0	0	0	0	0	0	0	0	0	0	20	0	0	0	0	0	0	0	0	0	0	0	0
n	0	0	0	0	0	0	0	0	0	0	0	0	0	20	0	0	0	0	0	0	0	0	0	0	0
o	0	0	0	0	0	0	0	0	0	0	0	0	0	0	20	0	0	0	0	0	0	0	0	0	0
p	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	20	0	0	0	0	0	0	0	0	0
q	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	20	0	0	0	0	0	0	0	0
r	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	20	0	0	0	0	0	0	0
s	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	19	0	0	0	0	0	0
t	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	20	0	0	0	0	0
u	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	20	0	0	0	0
v	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	20	0	0	0
w	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	20	0	0
x	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	20	0
y	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	20

Table (4.3) Confusion matrix of Multiclass Least Square Support Vector Machine

Number of gesture per class=20

Total class=25

Total no. gesture=20*25=500

Correctly classified gesture =496

Accuracy=$\left(\frac{correctly\ classfied\ gesture}{total\ no.of\ gesture}\right)\times 100\%$

$$= \left(\frac{496}{500}\right) \times 100 = 99.2\%$$

4.6 CONCLUSION

In this chapter we have discussed about different classification techniques. Local contour sequence which was determined in previous chapter has been used as input to different classfier.We have determined minimum distance between two classes in linear classifier .first we have calculated difference between our reference gesture and test gesture then based on that discrimination analysis we have assigned class to the test gesture. We have achieved 94.6% accuracy by using linear classifier. The second classification technique was support vector machine. Support vector machine is used in pattern recognition. It determine the optimal hyper plane between two class of data .we have used one-vs-all technique for multi class classification for our project. We achieved accuracy of 98.6% by using multiclass support vector machine. Then next we have used least square support vector machine as our classifier and achieved 99.2% accuracy.

REFERENCES

[1] Lalit Gupta and Suwei Ma "Gesture-Based Interaction and Communication: Automated Classification of Hand Gesture Contours" , *IEEE transactions on systems, man, and cybernetics—part c: applications and reviews*, vol. 31, no. 1, February 2001

[2] Shigeo Abe, "Support Vector Machines for Pattern Classification, second edition", Kobe University ,Graduate School of Engineering, 2nd edition, Springer-Verlag London Limited 2005, 2010

[3] Simon Haykin, "Neural Networks: A Comprehensive Foundation", 2nd edition Prentice Hall PTR ,1998

[4] O. L. Mangasarian, David R. Musicant, "Lagrangian Support Vector Machines", Journal of Machine Learning Research, 161-177 (2001)

[5] A. Smola and B. Schölkopf, "Sparse greedy matrix approximation for machine learning," in Proc. 17th Int. Conf. Mach. Learn., San Francisco, CA, pp. 911–918, 2000.

[6] Vikramaditya Jakkula,, "Tutorial on Support Vector Machine (SVM)" ,Vikramaditya Jakkula, School of EECS, Washington State University, Pullman, www.ccs.neu.edu/course/cs5100/resources/SVMTutorial.doc

[7] D. P. Wiens, "Designs for approximately linear regression: Two optimality properties of uniform designs," Stat. Probability Lett., vol. 12, pp. 217–221, 1991

[8] J. Weston, C. Watkins, "Multi-class support vector machines", Proceedings of ESANN99, Belgium, 1999.

[9] Liu Yucheng and Liu Yubin, "Incremental Learning Method of Least Squares Support Vector Machine", International Conference on Intelligent Computation Technology and Automation" VCL-94-104, 2010

[10] K. T. Fang, Y.Wang, and P. M. Bentler, "Some applications of numbertheoretic methods in statistics," Stat. Sci., vol. 9, pp. 416–428, 1994.

[11] http://asi.insa-rouen.fr/enseignants/~arakotom/toolbox/index.html

[12] http://www.esat.kuleuven.be/sista/lssvmlab

CHAPTER 5

CONCLUSION AND FUTURE WORK

5.1 CONCLUSION

In recent years a lot of research has been conducted in gesture recognition. The aim of this project was to develop an offline Gesture recognition system. We have shown in this project that offline gesture recognition system can be designed using SVM. It is determined that contour is very important feature and can be used for discrimination between two gesture. The processing steps to classify a gesture included gesture acquisition, segmentation, morphological filtering, contour representation and classification using different technique. The work was accomplished by training a set of feature set which is local contour sequence.

- Otsu algorithm is used for segmentation purpose and gray scale images is converted into binary image consisting hand or background .Morphological filtering techniques are used to remove noises from images so that we can get a smooth contour.
- We have used Local contour sequence as our prime feature. Canny edge detection technique is used to detect the border of hand in image. A contour tracking is applied to find the contour and pixel in contour is numbered sequentially. Local contour sequence for any arbitrary pixel is calculated as perpendicular distance from the chord connecting end points of window size w.
- The main advantage of LCS is that it is invariant to rotation ,translation and scaling so it is a good feature to train the learning machine as we have done with SVM and LSSVM .We have achieved 98.6% accuracy with SVM and 99.2% accuracy with LSSVM.

5.2 FUTURE WORK

- Area of Hand gesture based computer human interaction is very vast. This project recognizes hand gesture off-line so work can be done to do it for real time purpose. Hand recognition system can be useful in many fields like robotics, computer human interaction and so make this offline system for real time will be future work to do.
- Local contour sequence (LCS) is our prime feature in this project.LCS can be used with used with other features so that it can be optimize to achieve a higher recognition accuracy

- Support Vector Machine can be modified for reduction of complexity. Reduction of complexity leads us to a less computation time. Reduced complexity provides us less computation time so we can make system to work real time.